UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/690,643	10/23/2003	Yasuo Sugure	XA-9956	9148

181 7590 05/15/2006

MILES & STOCKBRIDGE PC
1751 PINNACLE DRIVE
SUITE 500
MCLEAN, VA 22102-3833

EXAMINER	
HUISMAN, DAVID J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 05/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/690,643	SUGURE ET AL.	
	Examiner	Art Unit	
	David J. Huisman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 October 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-15* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-15* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 October 2003* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *23 October 2003*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-15 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: IDS as received on 10/23/2003 and Foreign Priority Papers as receive don 3/12/2004.

Specification

3. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

4. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

5. The disclosure is objected to because of at least the following informalities:

- In the 2nd to last line on page 2, replace "shortens" with --shorten--
- The first sentence of the first full paragraph on page 17 is grammatically incorrect.

Appropriate correction is required and applicant is encouraged to try and locate any additional mistakes within the specification.

Drawings

6. The drawings are objected to because of the following minor informalities:

- In Fig.6, the examiner believes that "Total 8 States" be "Total 13 States". If the examiner is incorrect, the applicant is asked to confirm the correctness of the current figure.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Objections

7. Claim 6 is objected to because of the following informalities: Please replace "couple between" with --couple--. Also, the final paragraph is grammatically incorrect and difficult to understand, and consequently, it should be reworded. Appropriate correction is required.

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

9. Claims 1-6 are rejected under 35 U.S.C. 102(b) as being anticipated by Damron, U.S. Patent No. 6,108,767.

10. Referring to claim 1, Damron has taught a data processor using a status register and a plurality of register banks to execute instructions, wherein the status register includes an overflow flag to indicate an overflow of the plurality of register banks. See column 1, lines 37-49, and note the multiple register banks (register windows). Also, see Fig.7A, register 707, and note that that exception history is tracked, where "O" represents overflow of the register banks.

11. Referring to claim 2, Damron has taught a data processor comprising:

a) a status register. See Fig.7A, component 707.

b) a central processing unit (Fig.1) including a predetermined register set. See column 1, lines 22-28 and note that a first register window (first set of registers) is assigned to the program for use.

c) a plurality of register banks corresponding to the predetermined register set, wherein the plurality of register banks are used to save storage information held by the predetermined register set when an interrupt occurs. See column 1, lines 29-49, and note that a stack is used to store a current register window upon a subroutine call, which is a type of interrupt (as the program flow is interrupted). And, the stack may hold multiple register windows and so each window is stored in a stack bank (register bank)

d) wherein the status register includes an overflow flag to indicate an overflow of the plurality of register banks. See Fig.7A, register 707, and note that that exception history is tracked, where "O" represents overflow of the register banks.

12. Referring to claim 3, Damron has taught a data processing unit as described in claim 2. Damron has further taught that when an interrupt exception occurs in a state in which data has been saved to all banks of the register banks, and wherein when the accepted interrupt exception is permitted to use the register banks, the central processing unit saves data of the register set to a stack area and reflects an overflow state in the overflow flag. See column 1, lines 37-49, and note that when an overflow exception occurs (there are not enough registers to assign to a new subroutine), current registers must be spilled (saved) to a memory stack area and the corresponding bit in the status register (Fig.7, component 707) would be set to indicate overflow.

13. Referring to claim 4, Damron has taught a data processing unit as described in claim 3. Damron has further taught that when the overflow flag indicates an overflow state, if data restoration from the register banks to the register set is directed, the central processing unit restores the data from the stack area to the register set. See column 1, lines 45-57, and note that if overflow occurred (and registers were spilled to memory), then those registers are restored to

the register set for use. Specifically, when a subroutine needs registers that have been spilled, they will be brought back into the file for use.

14. Referring to claim 5, Damron has taught a data processing unit as described in claim 2. Damron has further taught that when an interrupt exception occurs in a state in which data has been saved to all banks of the register banks, and the accepted interrupt exception is permitted to use the register banks and specified to execute a predetermined exception handling routine, the central processing unit executes the predetermined exception handling routine and does not perform saving to the register banks. See Fig.3A, and note that the exception handling routine does not save anything to the register banks. Instead, registers are stored in memory.

15. Referring to claim 6, Damron has taught a data processing unit as described in claim 1. Damron has further taught:

a) a memory constituting the plurality of register banks. See column 1, lines 37-49 and note that register banks may be stored in memory when spilling occurs.

b) a bus dedicated to couple between the memory and a predetermined register set, wherein the bus includes as many bits as parallel data transfer is allowed in units of plurality of registers contained in the register set. Clearly, a bus must inherently exist between the memory and the registers so that they may be spilled to memory. And, no matter the bus size (16 bits, 32, buts, 64 bits, etc), the bus includes the same number of bits that is allowed for each transfer. That is, if the bus is 32 bits, then 32 bits or register information may be transferred in parallel.

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

17. Claims 7-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Damron.

18. Referring to claim 7, Damron has taught a data processing unit as described in claim 2. While Damron has taught that the central processing unit, in response to the occurrence of interrupt exception, saves information within the predetermined register set to the register banks (column 1, lines 37-49), Damron has not explicitly taught saving the status register and a program counter to a stack area. However, Official Notice is taken that storing status register contents and a PC value are just two things that are known to be stored upon some type of interrupt, exception, or subroutine call. These items are stored so that when the interrupt handler, for instance, is done executing, the system may be returned to the state that it was in before the interrupt occurred, which ensures correct program execution. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Damron to store a PC and status register in response to an interrupt.

19. Referring to claim 8, Damron has taught a data processing unit as described in claim 7. Damron has further taught that whether to save to the register banks is able to be selected according to factors indicating types of interrupts or priority levels. From column 1, lines 37-49, it can be seen that in response to underflow, saving to register banks occurs (registers are

restored), whereas, with other types of interrupts, such as an I/O device request, which would inherently exist, such saving does not have to occur.

20. Referring to claim 9, Damron has taught a data processing unit as described in claim 7. Damron has further taught interrupts to always perform saving to the register banks (underflow interrupts result in data being restored to the register banks) and interrupts capable of automatically selecting a stack area as a save location when the number of remaining banks is small. See Fig.3A and column 6, lines 24-40, and note that when the number of remaining banks is small (based on a prediction), the overflow interrupt will cause selection of a stack area in memory to hold one or more of the register windows in anticipation that more space will be needed.

21. Referring to claim 10, Damron has taught a data processing unit as described in claim 7. Damron has further taught that the central processing unit includes in an instruction set a register restore instruction to restore storage information from a register bank last saved to the predetermined register set. See column 1, lines 37-49.

22. Referring to claim 11, Damron has taught a data processing unit as described in claim 10. Damron has further taught that if the register restore instruction is executed when the register banks are empty, predetermined exception service occurs. See column 1, lines 37-57, and note that if the banks are empty, a register fill occurs.

23. Referring to claim 12, Damron has taught a data processing unit as described in claim 10. Damron has not explicitly taught that the instruction set includes a return instruction to restore a value of the program counter and a value of the status register stored in a stack area in interrupt exception handling and enable return to previous program execution processing. However,

Official Notice is taken that return instructions from subroutines and handlers are well known and expected in the art. Return instructions cause the program to return to the period just before the interrupt handler or subroutine was called, thereby allowing normal execution to continue. In order to determine the location where the program left off before calling the interrupt handler, the PC address must be known, and the status register must also be regenerated so that the program may execute based on the appropriate system status. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Damron such that the instruction set includes a return instruction to restore a value of the program counter and a value of the status register stored in a stack area in interrupt exception handling and enable return to previous program execution processing. Doing so would allow for correct returning from subroutines and handlers.

24. Referring to claim 13, Damron has taught a data processor that:

a) uses a predetermined register set and a plurality of register banks to execute instructions. See column 1, lines 22-49, and note that a current register window is the predetermined register set, and the plurality of register banks are used to hold previous register windows.

b) wherein the plurality of register banks are used to save storage information held by the predetermined register set. Again, the register banks hold register information that was once part of the predetermined register set (windows for old routines).

c) wherein an instruction set of the data processor includes a register restore instruction to restore storage information from a register bank last saved to the predetermined register set. See column 1, lines 37-49, and note that when a bank is spilled (saved) to memory, then it must eventually be restored using a restore instruction.

d) Damron has not explicitly taught that the instruction set includes a return instruction to restore a value of a program counter and a value of a status register saved to a stack area in interrupt exception handling and cause return to previous program execution processing. However, Official Notice is taken that return instructions from subroutines and handlers are well known and expected in the art. Return instructions cause the program to return to the portion of the program just before the interrupt handler or subroutine was called, thereby allowing normal execution to continue. In order to determine the location where the program left off before calling the interrupt handler, the PC address must be known (since the PC holds address of an instruction to be fetched), and the status register must also be regenerated so that the program may execute based on the appropriate system status. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Damron such that the instruction set includes a return instruction to restore a value of the program counter and a value of the status register stored in a stack area in interrupt exception handling and enable return to previous program execution processing. Doing so would allow for correct returning from subroutines and handlers.

25. Referring to claim 14, Damron has taught a data processor including:

a) a predetermined register set and a plurality of register banks corresponding to the predetermined register set, wherein the plurality of register banks are used to save storage information held by the predetermined register set. See column 1, lines 22-49, and note that a current register window is the predetermined register set, and the plurality of register banks are used to hold register windows.

b) wherein an instruction set of the central processing unit separately includes a register restore instruction to restore storage information from a register bank last saved to the predetermined register set. See column 1, lines 37-49, and note that when a bank is spilled (saved) to memory, then it must eventually be restored using a restore instruction.

c) Damron has not explicitly taught that the instruction set includes a return instruction to restore a value of a program counter and a value of a status register saved to a stack area in interrupt exception handling and cause return to previous program execution processing. However, Official Notice is taken that return instructions from subroutines and handlers are well known and expected in the art. Return instructions cause the program to return to the portion of the program just before the interrupt handler or subroutine was called, thereby allowing normal execution to continue. In order to determine the location where the program left off before calling the interrupt handler, the PC address must be known (since the PC holds address of an instruction to be fetched), and the status register must also be regenerated so that the program may execute based on the appropriate system status. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Damron such that the instruction set includes a return instruction to restore a value of the program counter and a value of the status register stored in a stack area in interrupt exception handling and enable return to previous program execution processing. Doing so would allow for correct returning from subroutines and handlers.

26. Referring to claim 15, Damron has taught a data processor as described in claim 14. Damron has further taught that:

a) in task switching using the interrupt exception handling, in return from interrupt exception handling, the register restore instruction is executed to restore data of a register bank in a task of a switching source to the register set, wherein the restored data is stored in an OS internal table managed by an OS, wherein register set data of a task of a switching destination is restored from the OS internal table to the predetermined register set, and wherein the return instruction is executed to transfer control to program execution processing of the task of the switching destination. When dealing with subroutines (as does Damron), when a particular subroutine is finished, a return instruction is executed to return to the portion of the program (another subroutine, if you will), which called the returning subroutine. As discussed in column 1, lines 22-49, each subroutine gets a register window and the current subroutine gets the current window (the predetermined set). When that subroutine is finished, the routine which it returns to is then the current subroutine and it must use its own registers. If its registers had been spilled to the OS table (memory), then they must be restored. Clearly, you must restore before you return because if it were the other way around, then the new subroutine would be accessing the wrong registers, which would cause incorrect execution.

Conclusion

27. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

Thimmanagari et al., U.S. Patent No. 7,024,541, has taught saving and restoring register windows and using multiple status registers to detect overflow/underflow.

Bui, U.S. Patent No. 6,487,630, has taught a processor with a register stack engine for dynamically spilling and filling registers to a backing store.

Hays et al., U.S. Patent No. 5,640,582, has taught register stacking in a task-switching system where overflow/underflow are trapped.

Weldon et al., "Quantitative Evaluation of the Register Stack Engine and Optimizations for Future Itanium Processors," Proceedings of the 6th Annual Workshop on Interaction Between Compilers and Computer Architectures, 2002, has taught general concepts such as spilling/filling registers and mapping frames to registers for current subroutines.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (571) 272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
April 26, 2006

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100